Method and arrangement for instruction word generation in the driving of functional units in a processor

Patent Claims

1. Method for the generation of instruction words for driving functional units in a processor, the instruction words comprising a plurality of instruction word parts and each instruction word part respectively driving a functional unit, in which, before a program sequence, a sequence of primary instruction words which originates from a translation of a program code undergoes fractionation into program words, and in which, during the program sequence, under the control of a program word which has an information part at least of the width of an instruction word part, an instruction word is taken from a row - determined by a reading row number - of an instruction word memory that can be written to row by row, the said instruction word is altered by means of substitution of an instruction word part by the information part of the respective program word and is written back to a row of the instruction word memory, the said row being determined by a writing row number, and after generation - effected in this way - of an instruction word corresponding to the primary instruction word to be executed, the said instruction word is output for driving the functional units, characterized in that the reading row number is provided by a read pointer register (18) and the writing row number is provided by a write pointer register (19), a reading and a writing row number being output per program word (17), and a number of successive reading and writing row numbers being determined by the content of a block length register (20).

2. Method according to Claim 1, characterized in that the program word (17) has a set bit (21) by which, given a set active state, the instruction word

generation (10) is interrupted and the register contents of the read pointer register (18), and/or of the write pointer register (19) and/or of the block length register (20) are set by the content of the information part of the program word and, given a set inactive state of the set bit (21), the generation of the instruction word (15) is carried out.

3. Method according to Claim 2, characterized in that, in the case of a program word (17) with a set active state of the set bit (21), the content of the information part is stored in the read pointer register (18), write pointer register (19) and block length register (20).

4. Method according to one of Claims 1 to 4 [sic], characterized in that the instruction word memory (24) is divided into a first instruction word memory page (6) and into a second instruction word memory page (7) each having the same row numbering, and in that, for the synthesis of the instruction word (15), the instruction word memory page to be called is determined by the content of a page register (27).

5. Method for the generation of instruction words for driving functional units in a processor, the instruction words comprising a plurality of instruction word parts and each instruction word part respectively driving a functional unit, in which, before a program sequence, a sequence of primary instruction words which originates from a translation of a program code undergoes fractionation into program words, and in which, during the program sequence, under the control of a program word which has an information part at least of the width of an instruction word part, an instruction word is taken from a row - determined by a reading row number - of an instruction word memory that can be written to row by row, the said instruction word is altered by means of substitution of an instruction

word part by the information part of the respective program word and is written back to a row of the instruction word memory, the said row being determined by a writing row number, and after generation - effected in this way - of an instruction word corresponding to the primary instruction word to be executed, the said instruction word is output for driving the functional units, characterized in that the instruction word memory (24) is divided into a first instruction word memory page (6) and into a second instruction word memory page (7) and in that, for the synthesis of the instruction word (15), the instruction word memory page to be called is determined by the content of a page register (27).

6. Method according to one of Claims 1 to 5, characterized in that an interrupt signal (25) immediately triggers, at the processor, during first task to processed [sic], on the first instruction word memory memory [sic] page (6) or the second instruction word memory page (7), buffer-storage of a left processing state of the first task on a global memory and then the execution of a second task on the unprocessed first instruction word memory memory [sic] page (6) or the second instruction word memory page (7), and in that, after the ending of the second task, after restorage from the global memory, the first task is continued in a manner rejoining the left processing state of the said first task.

7. Method according to one of Claims 1 to 6, characterized in that a prefetch unit (28) controls the set-up of the instruction word memory (24), and in that, independently of the processing state of the current task, the prefetch unit (28) provides an additional instruction word (29) in an unused row of the instruction word memory (24) or in an additional instruction word memory (30), if no new instruction

word (15) is obtained during the execution of a current task at the prefetch unit (28).

8. Arrangement for the generation of instruction words for driving functional units in a processor having functional units, having an instruction word memory assigned to these functional units, and having an instruction word buffer for storing already generated instruction words having a width which is at least equal to the bit width of the instruction word, the instruction word buffer comprising a memory with random or fixed-programmed row-by-row access, characterized in that the instruction word memory (24) is assigned a generation unit (31).

9. Arrangement according to Claim 8, characterized in that a block length register (20) a read pointer register (18) and a write pointer register (19) are arranged in the generation unit (31), the read pointer register (8) [sic] being assigned a read pointer up/down counter (32) and the write pointer register (19) being assigned a write pointer up/down counter (33), whose ring counting properties are determined by the content of the block length register (20).